

Autonomous Flight and Real-time Tracking of Unmanned Aerial Vehicle

Bogdan Muresan

Anglia Ruskin University

Faculty of Science and Technology

CB1 1PT, Cambridge, UK

Email:bogdan.Muresan@student.anglia.ac.uk

Shabnam Sadeghi Esfahlani

Anglia Ruskin University

Faculty of Science and Technology

CB1 1PT, Cambridge, UK

Email:shabnam.sadeghi-esfahlani@anglia.ac.uk

Abstract—This study describes a system in which a micro UAV (quadrotor) was coupled with a Kinect (v2), a Myo armband and an RGB camera. The quadrotor was connected to two PC clients or workstations and communicated through the Robot Operating System. The UAV moved to the marked targets in a cluttered environment without collision using the depth sensor. Recognises faces via the on-board camera based on the frame by frame basis and uses feature-based monocular simultaneous localisation and mapping (SLAM) in real-time. The SLAM tracks the pose of the quadrotor, simultaneously builds an incremental map of the surrounding environment to locate the UAV in that. The Myo armband was employed for teleoperation which commands the quadrotor to start/stop its journey or to begin a new task using hand gestures. The face recognition algorithm was developed using the Fisherface library and pre-trained database. Three missions were assigned to the UAV; to detect the marked area via Kinect's depth sensor, fly towards and hover around the marked area, send the image/video streams to the ground station and to look for the person's face in the crowded environment, match the name with the face owner and follow him/her within the distance of 2 m. Various organisations could use the proposed system for different purposes. It could be utilised for search and rescue, environmental monitoring, surveillance or inspection. It could be used to identify a person in a collapsed building, in urban/suburban areas or to locate people with a particular need (alzheimer or dementia casualties which leads to wandering behaviour).

Index Terms—Micro Unmanned Aerial Vehicle, Kinect v2, Myo armband, Monocular Simultaneous Localization and Mapping (SLAM).

I. INTRODUCTION

Deployment of intelligent robots has been made possible through technological advances in various fields such as artificial intelligence, robotics, wireless communication and control theories [1]. Intelligent robots can be embedded in many systems, especially to substitute human efforts in applications where a human-operation is dangerous, inefficient and impossible. For outdoor applications, various commercial systems are available based on GPS. However, the indoor application has not received much attention due to its complication with localisation and mapping. Such capabilities (the indoor applications) are critical, for instance, when involved in search and rescue missions, inside unprecedented fired, collapsed towers, natural disasters flood or an earthquake [2]. In a dynamically changing condition of environments, UAV

requires being equipped with various on-board sensors to be able to collect as many information as possible. Various sensors are used by several researchers to acquire the environments' specific information including relative distance from the ground surface or to detect objects [1], [3], [4], [5], [6], [7], [8], [9] and [10].

[9] equipped MAV (micro aerial vehicle) with a laser range-finder sensor, to explore and map unstructured environments and estimating MAV's position relatively. [11] used four cameras in parallel on a little rotary wing platform of MAV for autonomous and high-speed flight with onboard computer vision algorithms. [12] used monocular vision SLAM to stabilise the position of a quadrotor helicopter to build a low-resolution 3D map.

In this study, we proposed the use of Crazyflie (2.0) quadrotor which weighs 27 grams with limited payload. Our system facilitated a fast and an efficient orientation competence, accurate image-matching and face detection, localisation and mapping in a wide variety of cluttered unknown indoor environments. It was built in ROS package and inspired by the work of [8], [10] and [13]. The Kinect mapping tool provided the 3D depth information. The RGB camera facilitated face recognition, stream video and generated SLAM (simultaneous localisation and mapping) via computer vision algorithms. The Myo armband commanded the drone via hand gesture. The SLAM metrically closed loops in real-time and reused the map in already mapped areas for fast localisation [14]. The results showed that the quadrotor could stably deliver defined tasks, access the trained database and follow the face owner. It could robustly hover, take off, land and fly towards marked trajectories.

II. DESIGN AND METHODOLOGY

A. Kinect v2 and Myo armband

The Microsoft KinectTM device consists of an infrared laser (IR emitter) and an RGB-D camera for depth sensing [15]. The internal parameters of the RGB-D camera of the Kinect were calibrated using a chessboard pattern. It visualises the environment to identify the reference frames and positions of the quadrotor and the target (colour marked locations). Kinect has some limitations for obstacle avoidance due to its fixed location, heavyweight, lack of portability and narrow imaging

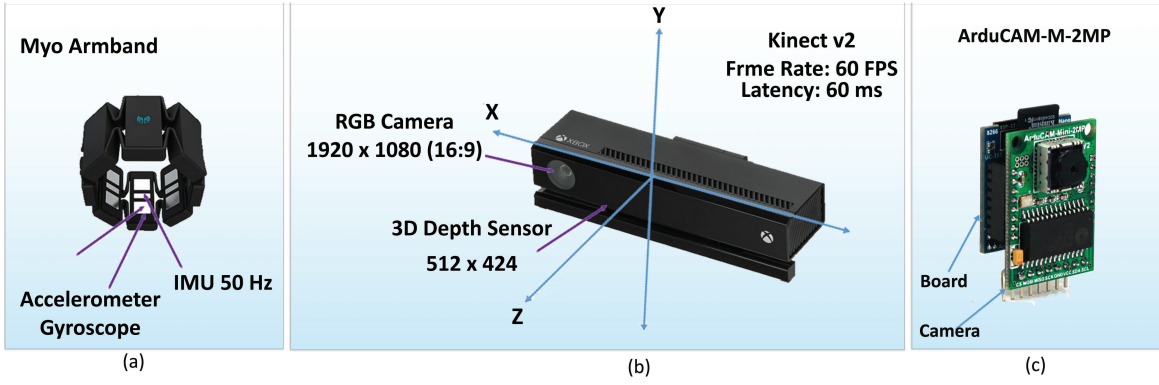


Fig. 1. (a) The Myo armband, (b) Kinect v2, and (c) Arducam camera with an Arduino board.

area (about 58° horizontal and 43° vertical).

The teleoperating Myo armband device¹ used journey via hand gestures (Double-tap to start and Fist to stop it). The orientation of an arm was also mapped into Myo armband to pilot the MAV if required and the mapping is listed Table I. Myo is connected via Bluetooth USB adapter to the work station to record and collect real-time orientation and hand gestures. Myo armband is shown in Fig. 1-a, RGB-D (Kinect) illustrated in Fig. 1-b and RGB Arducam is in Fig. 1-c).

B. Crazyflie (2.0) Quadrotor

The Crazyflie (2.0)² is classified as a micro air vehicle (MAV). It is depicted in Fig. 3-a with the Arducam and a battery. The quadrotor has two pairs of counter-rotating propellers (four fixed-pitch propellers). These propellers are organised in opposing pairs and rotate in opposite directions. One pair rotate clockwise, while the other two rotate counter-clockwise. This way the torque generated by one pair is cancelled by the other, offering a neutral sum of rotational forces to keep the quadrotor stable. They are counter-rotating to negate the torque that would cause the body of the quadrotor to rotate in the opposite direction if only one propeller were turning. While hovering, the propellers rotate at the same speed and provide the lift to keep the quadrotor in the air, overcoming the pull of gravity. As a consequence, manipulating the speed difference between the pairs of rotors generates a torque that determines the rotation of the UAV in the horizontal plane. By adjusting the speed of various propellers, the quadrotor can obtain a whole range of motions, from rising/falling to back/front/left/right tilt. Fig. 2 illustrates the top view of the quadrotor and the way various orientations could be enabled. The quadrotor's micro-controllers are; ARM Cortex-M4 embedded processor (STM32F405) and Nordic Semiconductor (nRF51822) [17], [18] and [19]. The processor runs at 168 MHz and reads data from 10-DOF IMU with nine-axis of (accelerometer, gyroscope and magnetometer) and a high precision pressure sensor [20] and [21]. The nRF51822 handles power management and the radio communication.

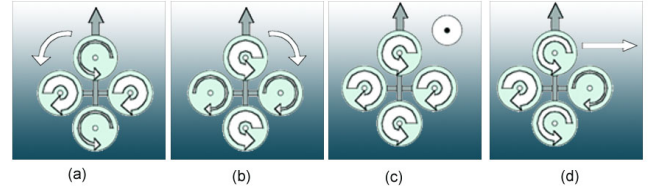


Fig. 2. Quadrotor dynamics (a) rotate-left, (b) rotate-right (difference in torque manipulates the yaw angle), (c) hovering and vertical thrust (balanced torque to go up/down) and (d) difference in thrust to handle the pitch and roll angle (move-right) [16].

The Bluetooth Low Energy (BLE) and Compressed Real-Time Protocol (CRTP) are also supported by the nRF51.

The conventional pinhole-camera theory was used for the camera calibration as shown in Fig. 3-b and Fig. 3-c. The world was modelled as a 3D projective space and the model for the image was the 2D projective plane. The central projection of points in space onto a plane, the centre of projection be the origin of Euclidean coordinate system and the plane $Z = f$ where f is the focal plane. Under the pinhole camera model, a point in space with coordinate $X = (X, Y, Z)^T$ is mapped to the point $x = (fX/Z, fY/Z, f)^T$ on the image plane as illustrated in Fig. 3-b and Fig. 3-c. If $X_{4 \times 1} = (X, Y, Z, 1)$ be the world point, x_3 be the image point and the camera matrix P for the pinhole model of central projection be $x = PX$ which shows the mapping from Euclidean \mathbb{R}^3 to \mathbb{R}^2 as illustrated in Eqn 1. Thus, $P = \text{diag}(f, f, 1) [I_{3 \times 3} \mid \mathbf{0}_3]$ is the camera matrix and the projective transformation is as follows:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = P_{3 \times 4} \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix} \quad (1)$$

If the origin of coordinates been transferred to a principle point as $(p_x, p_y)^T$ then

$$(X, Y, Z, 1)^T \mapsto (fX + Zp_x, fY + Zp_y, Z)^T \quad (2)$$

¹<https://www.thalmic.com/team/>

²<https://www.bitcraze.io/crazyflie-2/>

TABLE I
THE ORIENTATION OF AN ARM IS MAPPED INTO MYO ARMBAND TO PILOT THE MAV.

MAV	Arm/Myo
Θ : Pitch, (depth)	Roll
φ : Trust, (up-down)	Pitch
Φ : Roll, (left-right)	Yaw

It is equivalent to;

$$\begin{bmatrix} f & p_x & 0 \\ & f & p_y \\ & & 1 \end{bmatrix} (X, Y, Z, 1)^T \quad (3)$$

$$x = \begin{bmatrix} f & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} [I \mid 0] X_{cam} = k [I \mid 0] X_{cam} \quad (4)$$

Where k is the camera calibration matrix. Points in the

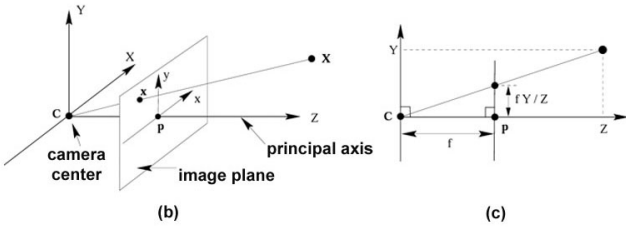
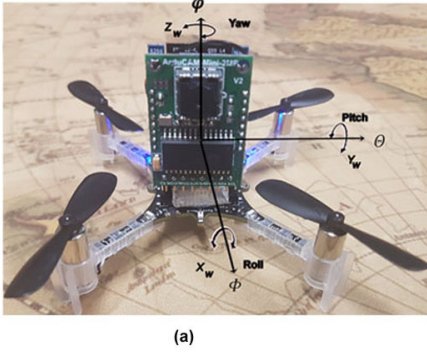


Fig. 3. (a) The Crazyflie (2.0) with mini ArduCAM, (b) and (c) Pinhole camera geometry where c is the camera centre and placed at the coordinate origin and the image plane is placed in front of the camera with P the principle point [22].

world coordinate frame with two coordinates are related via a rotation and a translation; $X_{cam} = R(X - \tilde{C})$. \tilde{C} is the camera coordinate centre in the world coordinate frame and the general mapping of a camera with 9 DOF can be written as:

$$P = k [R \mid -R\tilde{C}] \quad \text{or} \quad X_{cam} = RX - R\tilde{C} \quad (5)$$

The Euclidean geometry of the 3D world is determined by specifying a particular plane in P^3 . Euclidean transformation (translation and rotation) model, the motion of a rigid object can be written as Eqn 6. To determine the relative rotation, translation and time offset between the camera and the quadrotor we perform some calibration at regular time

intervals t_k . The transformation between the camera and the quadrotor is $T_{c,i}$ the pose of the quadrotor is $T_{w,i}(t)$ and the accelerometer and gyroscope have biases. The transformation that takes points from the quadrotor's IMU frame to the world frame at any time t can be written as Eqn. 6 [23];

$$T_{w,i}(t) = \begin{bmatrix} C(\Phi_\varphi(t)c_\varphi) & T(t) \\ 0^T & 1 \end{bmatrix} \quad (6)$$

where $C(\Phi_\varphi(t)c_\varphi)$ represents a rotation matrix from orientation parameters, $T(t)$ is the translation from which the velocity and acceleration could be calculated [24]. Encountered error in estimating time offsets from measurement data can be calculated by:

$$e_j = y_j - h(\Phi(t_j + g)c) \quad (7)$$

Where y_j is a measurement received at t_j time-stamp, $h(\cdot)$ is a measurement model that produces a predicted measurement from $x(\cdot)$ [25].

C. Feature Descriptor

Oriented FAST and Rotated BRIEF (ORB) is a fast binary descriptor with rotation invariant and fewer image noises. ORB-SLAM was proposed by [26] and built upon the well-known FAST (Features from Accelerated Segment Test) that match visual features such as Parallel Tracking and real-time keypoint detector [27] and the BRIEF (Binary Robust Independent Elementary Features) descriptor [28]. Feature-based monocular ORB-SLAM descriptor operates in real-time which is computationally efficient compared to FAST and BRIEF. It extracts features in less than 33 ms per image. ORB has a good invariance to a viewpoint which allows to match them from wide baselines, boosting the accuracy. The system localises the camera, builds the new map and tries to close loops. Fig. 4-a illustrates the system set up in an indoor environment with Crazyflie, Myo armband and Kinect v2. In this scenario, the Crazyflies' orientation is controlled by the Myo. Fig. 4-b and Fig. 4-c show the SLAMs generated by the RGB camera. Both show the camera's location, KeyFrames, local mapping (in red and black dots), loop closure and Covisibility. The Spanning Trees (green lines) are followed by a loop closure to correct and adjust deviations and drifts. The tracking is in charge of localising the camera in every frame and insert a new keyframe as required to provide a fitness approach. Camera localisation improves tracking robustness and enhances operation by discarding redundant keyframes [29].

The bag-of-words technique uses a visual vocabulary to convert an image into a scattered numerical vector, and to create an image database [30] and [31]. The visual vocabulary is

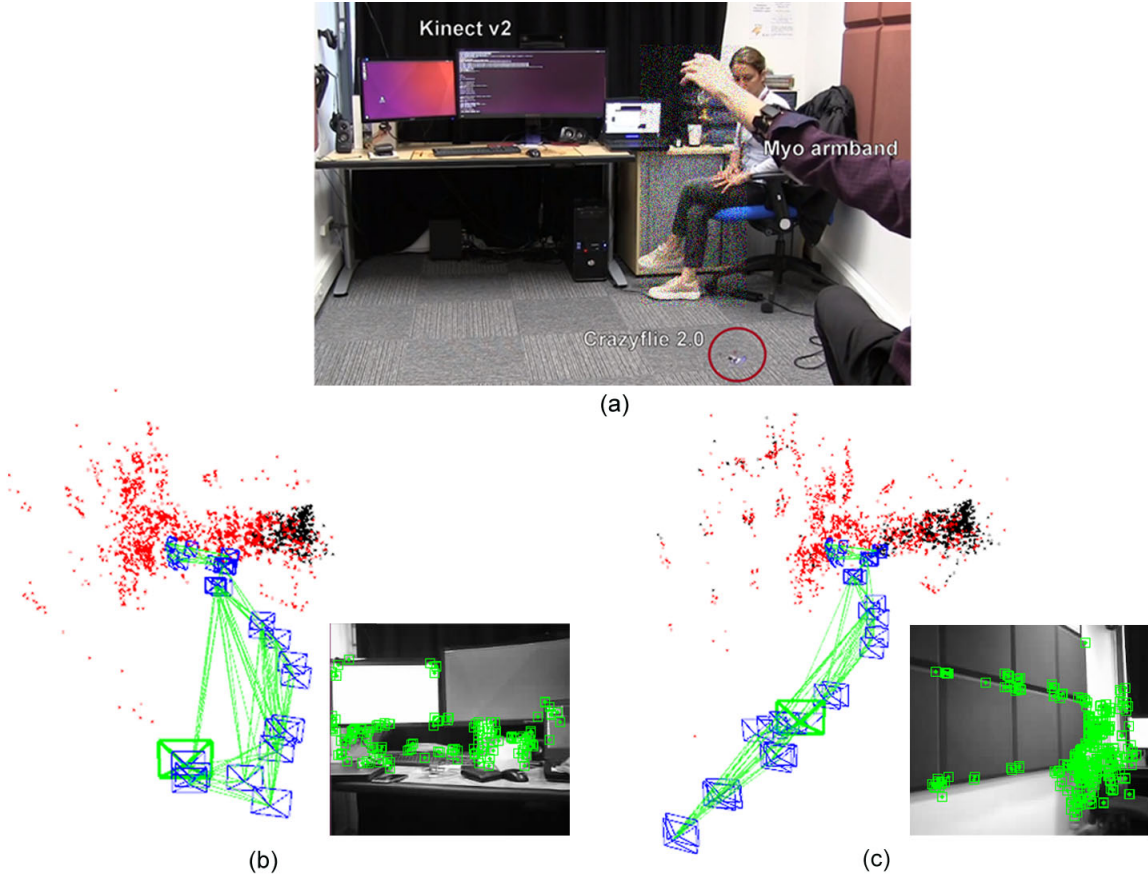


Fig. 4. (a) The system arrangement in an indoor environment to command the Crazyflie to start the journey using the Myo and the Kinect for depth sensing.(b) and (c) are the trajectories and sparse reconstruction of the sequences with multiple loop closures.

created offline by discretising the descriptor space into different visual words that helps with detecting revisited places. DBoW1/2 proposed by [32] are binary features that use FAST and BRIEF features [33]. For each word presented in the image, there is a score where a higher score represents a higher frequency of a word in the image, and a lower score is a word's lower frequency in the training dataset. This bag-of-words vector is compared against the images in the database using an inverted index. The similarity between two bag of word vectors v_1 and v_2 is the L_1 -score: $s(v_1, v_2)$, whose value lies in $[0, \dots, 1]$. This score is normalised, where the score of one means an image is a previously visited place.

$$\eta(v_1, v_2) = \frac{s(v_i, v_j)}{s(v_i, v_{i-1})} \quad (8)$$

The database is searched initially for individuals with high scores (a threshold α). The group with the highest score is considered as a loop candidate for the query image. The groups with the highest scores for the last k images must form an overlapping sequence where the consistency check improves the robustness of the system [34].

D. Face Detection & Recognition

Image matching is a fundamental aspect of many computer vision problems including face, object or scene recognition

[35], [36], [37] and [38]. In this study, a continuous face recognition is performed using Fisherface method in OpenCV library through ROS framework [39] and [40]. Fishers Linear Discriminant (FLD) [41] is a class-specific method that tries to shape the scatter to make it more reliable for classification. W_{opt} is selected in a way that the ratio of the between-class scatter and the within-class scatter is maximised [40] as defined in the equation below;

$$W_{opt} = \arg \min_W \frac{|W^T S_B W|}{|W^T S_W W|} = [w_1 \ w_2 \ \dots \ w_m] \quad (9)$$

where $\{w_i | i = 1, 2, \dots, m\}$ is the set of generalised eigenvectors of S_B and S_W corresponding to the m largest generalized eigenvalues $\{\lambda | i = 1, 2, \dots, m\}$ as;

$$S_B w_i = \lambda_i S_W w_i \quad (10)$$

To reduce the within-class scatter while preserving between-class scatter, W is picked to maximise the between-class scatter of the projected samples after having first reduced the within-class scatter. Taken to an extreme [40], maximises the between-class scatter of the projected samples subject to the constraint that the within-class scatter is zero, as such;

$$W_{opt} = \arg \min_{W \in W} |W^T S_B W| \quad (11)$$

W is the set of $n \times m$ matrices with orthonormal columns contained in the kernel of S_W .

Fig. 5-a illustrates the ROS client and its interfaces with the Arducam for video streaming. The data are saved by the client to visualise the mapping and localisation of the quadrotors' trajectories. The master is used to store topics and services registration information for nodes and to communicate this information to master. The UAV is programmed to identify faces based on the training database to match the faces to the names, follow them within a specific distance. Fig. 5-b shows a subject that is detected by the camera with a red square around the face and the distance of the camera to the subject is around 2m.

E. Bundle Adjustment

Bundle Adjustment (BA) is the method of refining a visual reconstruction to produce optimal 3D structure and viewing parameter camera pose and calibration) estimates. The parameters are estimated by minimising some cost functions that quantify the model fitting error. At the same time, the solution is simultaneously optimised regarding both structure and camera variations [42]. The system performs BA to optimise the camera pose in the tracking thread, optimise keyframes and points in the mapping thread, and uses a loop closure to optimise all keyframes and points. If the tracking is lost (e.g. due to sudden movement or connection loss) the place recognition module is used to perform a global relocalisation. Once there is an initial estimation of the camera pose and feature matchings, a local map is retrieved managing the visibility graph of keyframes. Real-time loop closing is based on the optimisation of a pose graph that is built from a spanning tree and links edges.

One of the limitations of our study was quadrotor's payload capacity and the battery charge which limited the flight time to 15 min in each journey which will be addressed in the future study.

III. RESULTS & DISCUSSION

A. Results

In this study, the RGB-D and RGB cameras were interfaced with the robotic operating system using ROS communication practice. The marked areas were defined via masks around the shaded areas using Kinect. It was implemented to coordinate the location of the quadrotor/the targets, to visualise the unknown environment, to generate the map and to position the quadrotor in the map. We adjusted the face detection and hovering of UAV at the marked locations via the messages and nodes that identify the reference frames and targets. We performed the place recognition based on the tracked and mapped features, the robust frame-rate relocalisation and loop detection. The tracking localised the camera within frames and decided when to insert a new keyframe. We used Myo for teleoperation and to send commands to the Crazyflie (double tap and fist) to start/stop as well as arm/hand flexion/abduction/adduction and extension for orientation.

IV. DISCUSSION

In this study, the algorithms were developed to enable real-time face and hand gesture recognition, as well as to generate SLAM in an indoor environment. The system could accurately detect faces, fly autonomously and avoid obstacles in an unknown environment. This idea could be utilised in urban search and rescue (S&R) missions. S&R is a multi-disciplinary activity due to the specialised nature of the work and consists of teams include personnel from police, firefighters and emergency medical services. Incorporating the proposed system in these situations could help to provide a training environment. The future work will expand the project to adopt a UAV with a higher payload to test the algorithm in urban search and rescue for training purposes. Development of this system allows a human operator to be trained using a virtual environment, use this system for robot/human interaction, to explore an environment and report to the ground station based on collected data, ensuring that full coverage of the case has been obtained.

REFERENCES

- [1] H. J. Kim and D. H. Shim, "A flight control system for aerial robots: algorithms and experiments," *Control engineering practice*, vol. 11, no. 12, pp. 1389–1400, 2003.
- [2] T. B. Glass, *A survey of usar healthcare practitioners' requirements in order to operate effectively in the collapsed structure environment*. PhD thesis, 2016.
- [3] W. Giernacki, M. Skwierczyński, W. Witwicki, P. Wroński, and P. Kozierski, "Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering," in *Methods and Models in Automation and Robotics (MMAR), 2017 22nd International Conference on*, pp. 37–42, IEEE, 2017.
- [4] L. Campos-Macías, D. Gómez-Gutiérrez, R. Aldana-López, R. de la Guardia, and J. I. Parra-Vilchis, "A hybrid method for online trajectory planning of mobile robots in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 935–942, 2017. <http://ieeexplore.ieee.org/document/7822897/>.
- [5] S. Tijmons, G. C. de Croon, B. D. Remes, C. De Wagter, and M. Mulder, "Obstacle avoidance strategy using onboard stereo vision on a flapping wing mav," *IEEE Transactions on Robotics*, 2017.
- [6] R. He, A. Bachrach, M. Achtelik, A. Gerafard, D. Gurdan, S. Prentice, J. Stumpf, and N. Roy, "On the design and use of a micro air vehicle to track and avoid adversaries," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 529–546, 2010.
- [7] J. Engel, J. Sturm, and D. Cremers, "Scale-aware navigation of a low-cost quadcopter with a monocular camera," *Robotics and Autonomous Systems*, vol. 62, no. 11, pp. 1646–1656, 2014.
- [8] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an rgb-d camera," in *Robotics Research*, pp. 235–252, Springer, 2017.
- [9] A. Bachrach, S. Prentice, R. He, and N. Roy, "Range-robust autonomous navigation in gps-denied environments," *Journal of Field Robotics*, vol. 28, no. 5, pp. 644–666, 2011.
- [10] E. S. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *The International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, 2011.
- [11] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A system for autonomous flight using onboard computer vision," in *Robotics and automation (ICRA), 2011 IEEE international conference on*, pp. 2992–2997, IEEE, 2011.
- [12] S. Ahrens, D. Levine, G. Andrews, and J. P. How, "Vision-based guidance and control of a hovering vehicle in unknown, gps-denied environments," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 2643–2648, IEEE, 2009.
- [13] C. Fairchild and T. L. Harman, *ROS Robotics By Example*. Packt Publishing Ltd, 2016.

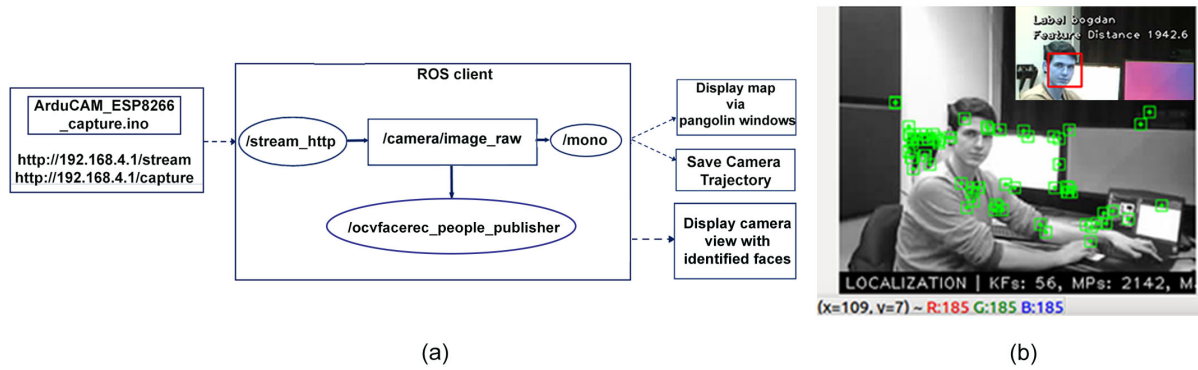


Fig. 5. (a) ROS client with camera interface for SLAM and face recognition nodes and messages. (b) The places (features) are detected via parallax (green dots) and face is detected via a red square and labeled by the name.

- [14] N. Kejriwal, S. Kumar, and T. Shibata, "High performance loop closure detection using bag of word pairs," *Robotics and Autonomous Systems*, vol. 77, pp. 55–65, 2016.
- [15] L. Yang, L. Zhang, H. Dong, A. Alelaiwi, and A. El Saddik, "Evaluating and improving the depth accuracy of kinect for windows v2," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4275–4285, 2015. <http://ieeexplore.ieee.org/abstract/document/7067384/>.
- [16] Y. Naidoo, R. Stopforth, and G. Bright, "Quad-rotor unmanned aerial vehicle helicopter modelling & control," *International Journal of Advanced Robotic Systems*, vol. 8, no. 4, p. 45, 2011.
- [17] B. Galea, E. Kia, N. Aird, and P. G. Kry, "Stippling with aerial robots," in *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering*, pp. 125–134, Eurographics Association, 2016. <http://www.cs.mcgill.ca/~kry/pubs/stippling/stippling.pdf>.
- [18] J. Engel, J. Sturm, and D. Cremers, "Camera-based navigation of a low-cost quadcopter," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 2815–2821, IEEE, 2012.
- [19] R. Carrera and R. Khanna, "3d scene reconstruction with a mobile camera,"
- [20] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*, pp. 649–666, Springer, 2016.
- [21] A. Bry, C. Richter, A. Bachrach, and N. Roy, "Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments," *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 969–1002, 2015.
- [22] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [23] R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An introduction to splines for use in computer graphics and geometric modeling*. Morgan Kaufmann, 1987.
- [24] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 1280–1286, IEEE, 2013.
- [25] P. C. Hughes, *Spacecraft attitude dynamics*. Courier Corporation, 2012.
- [26] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE international conference on*, pp. 2564–2571, IEEE, 2011.
- [27] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pp. 225–234, IEEE, 2007.
- [28] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," *Computer Vision–ECCV 2010*, pp. 778–792, 2010.
- [29] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [30] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *null*, p. 1470, IEEE, 2003.
- [31] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, vol. 2, pp. 2161–2168, Ieee, 2006.
- [32] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.
- [33] R. Mur-Artal and J. D. Tardós, "Fast relocalisation and loop closing in keyframe-based slam," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 846–853, IEEE, 2014.
- [34] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [35] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [36] P. Wagner, "Face recognition with opencv," *OpenCV 2.4. 9.0 Documentation*, 2012.
- [37] F. Lier, M. Hanheide, L. Natale, S. Schulz, J. Weisz, S. Wachsmuth, and S. Wrede, "Towards automated system and experiment reproduction in robotics," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pp. 3298–3305, IEEE, 2016.
- [38] D. Anifantis, E. Dermatas, and G. Kokkinakis, "A neural network method for accurate face detection on arbitrary images," in *Electronics, Circuits and Systems, 1999. Proceedings of ICECS'99. The 6th IEEE International Conference on*, vol. 1, pp. 109–112, IEEE, 1999.
- [39] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pp. 586–591, IEEE, 1991.
- [40] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 7, pp. 711–720, 1997.
- [41] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of human genetics*, vol. 7, no. 2, pp. 179–188, 1936.
- [42] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment a modern synthesis," in *International workshop on vision algorithms*, pp. 298–372, Springer, 1999.